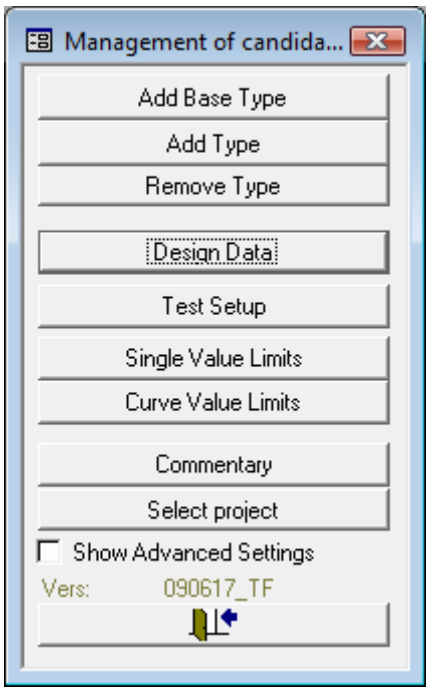
Using the Tas Parameter Database



Discom GmbH
Neustadt 10-12
37073 Göttingen
Germany
www.discom.de

Version 2.1

25.6.2009

Content

Parameter Database **4**
General Remarks 4
Parameter Cache 4
Two database files are used for the parameter setup 4
Open the parameter management 5
The initial dialog 5
Fundamental terms 5
General Handling 8
Data Organization in Lists 8
General Functions 9
Base Types and Bench Groups 9
Using the Key Selection Fields 9
Changing all entries of a Column 9
Sort data sets 10
Copy, Print and Compare 10
Type management: Create and Remove a Type 11
New Type or New Basetype? 11
Creating a New Base Type 11
Creating a New Type 12
Deleting Types and Base Types 13
Changing Teeth Numbers 14
Test Setup and Learning 16
Four Lists of Test Parameters 16
General Settings for the Learning Process 16
Setting Limits 18
Limits for Single Values 18
Limits for Curve Values 19
Measurements and Their Meaning 21
Introduction 21
The Standard Path of Processing 21
Sync Processing and Mix Processing 21
The Unit „Order“ 22
The Measurement Peak 22
The Measurement Rms 22
The Measurement Crest 22
The Measurement Kurtosis 22

The Measurement Spectrum 23
The Measurement Spectral Value 23
Track Measurements 23
Measurements Which are Determined at the End 24
Defining Measurements 25
General Measurement Parameters 25
Adding Measurements 26
Deleting Measurements 26
Interaction between List of Measurements and Limit Settings 27
Adding New Measurements 27
Security and Maintainance Issues 28
Database-Backup 28
De-fragmenting the database 28
External Database Modifications 29

Parameter Database

General Remarks

Most of the parameters for the Tas-system are set in the parameter database. The Tas system can not be used without a parameter database. This is why it is important to be familiar with the basic concepts of this database system.

Parameter Cache

The measurement program does not access the parameter database directly during the measurement. Instead, it uses data which is stored in the parameter cache. The way how the parameter cache gets its data from the parameter database can differ, depending on whether a central parameter server is used or not.

If the parameter database is located on a central server, the cache data files are usually generated on this central server either. The service program which does this job is called the “database cacher”. It reads out the parameter database in regular intervals, generates new cache data files, if necessary and copies them to the measurement PCs via network as soon as possible.

If no central server is used for cache file generation, the cache files are generated by the measurement program itself. This is done at the moment when the test bench control indicates that a new test for type X shall be carried out. The measurement program checks whether parameters for type X have been changed since the cache file for this type was created. If yes, it updates the cache file with data from the database. This update process may take some time, depending on the amount of data in the database. If no update is necessary, the existing cache file will be used and the measurement program is prepared for testing immediately.

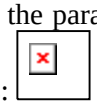
Two database files are used for the parameter setup

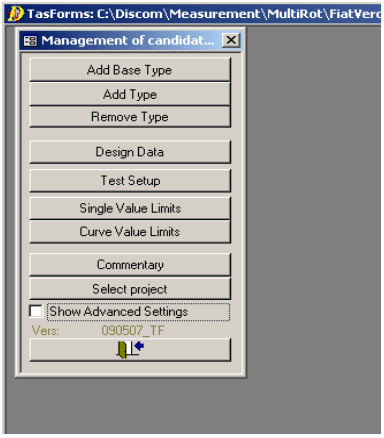
Independent of the place where the parameter database is stored (on a central server or on the measurement PC), two database files are used for the parameter setup. Currently, both files are Access 2000 database files (suffix .mdb). The first database file which ends on “QDb.mdb” holds the parameter data. This database file is being read out when the parameter cache data files are generated. The second database file (TasForms.mdb) holds the user interface program which allows to change parameters in the first database file. There are no measurement parameters stored in the user interface program. Instead, the tables of the first database file are linked into the user interface program. Otherwise, editing of parameters would be impossible.

In the following chapters I describe how to handle this user interface program, in the following referred to as „parameter management“.

Open the parameter management

The initial dialog

You open the parameter management with a link which usually shows a yellow „D“ as follows: 



You find this link on the desktop or in a folder on the desktop „Rotas for Experts“. A double-click on the link opens the parameter management. It shows up with the initial dialog shown on the right:

The title row of the main window shows the path where the current parameter database file is located. The sub-window contains buttons which mostly open further windows where you can edit certain data sets. Above the lowest button, with which you can close the program, it shows a version number. In case of questions concerning the parameter management, it is useful to know the version number of the program. There may exist newer versions of the program with slightly different functionality.

The check box „Show Advanced Settings“ expands the list of buttons. At the moment, these further buttons are skipped since you need advanced knowledge if you want to use them. You will read more about them, later.

Fundamental terms

The parameter management deals with things of different origin. Some things originate from the technical organization of databases, other things originate from transmission design, further other things originate from scientific acoustics. In the following the most important terms which are used in the parameter management will be briefly explained.

Data set: A data set is a collection of different entries in a database which belong together in some way. Each data set is uniquely identified by a **key**. If you know the key to a data set, the database can easily find the corresponding data. If you provide a complete key, there can be only one data set in the database for that key.

Type/ Base type: The objects (mostly transmissions) which shall be analyzed in a test, are referred to as *types* in the parameter management. To be more exact, the types are the names with which the test bench control announces a certain object (e.g. a transmission) to the measurement program. Each type belongs to a *base type*. The base type uniquely identifies the data set. The reason for this distinction is the following: If a transmission is being produced for some time, you gain experience about the transmission which may result in changes. Sometimes, only slight modifications in the housing are being done. Since it is necessary to distinguish the changed transmission from the original one, a new name is given to the changed transmission. For the parameter management, the changed transmission is a new *type*. But since slight modifications may not result in a changed acoustical behavior, the new type is assigned to the old *base type*. This has the effect that the new type uses exactly the same data set as the old type.

Test bench/ Bench group: The idea described above to access one type data set with different „names“ has been transferred to test benches also. I this context, each *bench*

group stands for a data set which can be used for different *test benches*. The effect is the same as above: Each test bench of a bench group uses exactly the same data set.

Model variant: Some test benches are able to test different transmission variants, e.g. transmissions having a front drive and 4 wheel drive variant. In the parameter management, this difference is covered by *model variants* if it results in differences for the parameters. Furthermore, the model variant defines the calculation model for the relative frequencies for the analysis. In general, the parameter database can even hold parameters for very different transmissions.

Test state: The different possible steps of a test run are represented by test steps in the parameter management. Usually, a transmission is switched once into each gear and then tested with positive torque ("Drive") and with negative torque ("Coast") in each gear. This results in test steps with names „1-D" (1st gear drive) or „6-C" (6th gear coast for example.

Location/ Wheel/ Rotor: A location in terms of the parameter management means a certain part which shall be analyzed. Transmissions, for example, contain cogwheels, shafts and bearings especially. The gear meshes of cogwheels are the major noise source of a transmission. Therefore, they can be accessed as *wheels* in the parameter management. Wheels which run with different speed can be analyzed using rotor synchronous analysis. Often more than one wheel in a transmission rotates with the same frequency, especially if they are fixed on the same shaft. The parameter management distinguishes this situation and uses the term *rotor*. For each frequency in the system, there shall be only one rotor. This avoids double analysis and wrong conclusions. Single wheels may be rotors, sometimes. Usually, shafts are rotors. Under certain circumstances (disadvantageous transmission ratio) even wheels can combine to a rotor if they have no direct connection (this means, that they are not connected to the same shaft).

Re-sampling: At the beginning of the noise analysis process you have the recording and digital re-coding of the noise. After the re-coding, you get blocks of digital data. The resolution of a block can vary, depending on the goal of the analysis. Rotor-synchronous analysis usually sets the resolution in a way, that each block contains multitudes of revolutions of a certain rotor (see above). The further processing (averaging, FFT, maximizing) is being done on the basis of these blocks.

Sensors/ Control values: Some external signals are necessary to do noise analyses. At first, you need the control signals from the test bench (information about the object to test and the current test step). Then you need speed information and of course, the noise signal. When processing, these signals have different roles. While noise signals (which can be recorded by more than one *sensor*) are re-coded during the re-sampling process (see above), the speed information is used as a *control value* ("when has one revolution of the part been completed"). Other control values are time or maybe a torque signal. Sometimes you may even analyze one control value relative to another control value, e.g. a torque track relative to changing speed or a speed track relative to time.

Measuring/ Triggering: Usually, the noise analysis is not being done permanently. Rotor synchronous analysis is useless at moments when the transmission changes gears (in contrast, gear shift analysis for manual gear boxes can be done at these moments). The different analysis tasks are distinguished by *triggers* in the measurement program. Triggers define which start/stop criteria is relevant for certain measurement values. The start/stop criteria itself can be of different kind. The simplest way to trigger a measurement is when the test bench control gives specific commands to start and stop a measurement (usually this is the case for gear shift analysis). The other way is to use control values (see above) as references for starting and stopping a measurement. This means for example that a measurement is started

when the speed passes a value X and that it is stopped when the speed passes a value Y. Some measurements are even impossible without control values as references. Any tracking over speed needs beside start and stop also information about the resolution of the track (take one track value for each X rpm).

Instruments/ Instrument Parameters: In terms of the parameter management the different analysis tasks are being done by different *instruments*. An instrument stands for a certain way of analyzing a signal or the result of such an analysis. The instrument "Peak" for example stands for the result of an analysis where just the maximum value of a data block is being taken. With *instrument parameters* you can define further parameters which are relevant for measuring a certain value. One important parameter for example is the trigger (see above) which gives the start and stop command for measuring a certain value.

Channel: In terms of the parameter management, the *channel* distinguishes the evaluation channel of a signal. Three typical channels are **Sync**, **Mix** and **Fix**. The channel Fix stands for an analysis where the signals are being sampled with fixed frequency (e.g. for signals of gear shift analysis). Both Mix and Synch are based on rotor-synchronous sampling. The difference between the two analysis channels is that rotor-asynchronous content is being mostly suppressed in the channel Synch.

Measurements: A measurement in terms of the parameter management is an analysis being done for certain locations and sensors in different channels. It depends on the test step whether an analysis is possible or not (some locations can not be analyzed in all test steps). Most instruments are able to carry out more than one analysis at a time. They are distinguished by instrument parameters, then. The result of a measurement can be of different kind: Single values, curves or curve sets (e.g. a spectrogram).

Limits/ Learning: Measurements can be evaluated by comparing them with limits. Since measurements can produce different types of results, the limits must also be of different types. Basically, all limits are learned limits. That means, they are built from mean value and standard deviation collected over as much measurements as possible using rules which are defined in the parameter database (details see below).

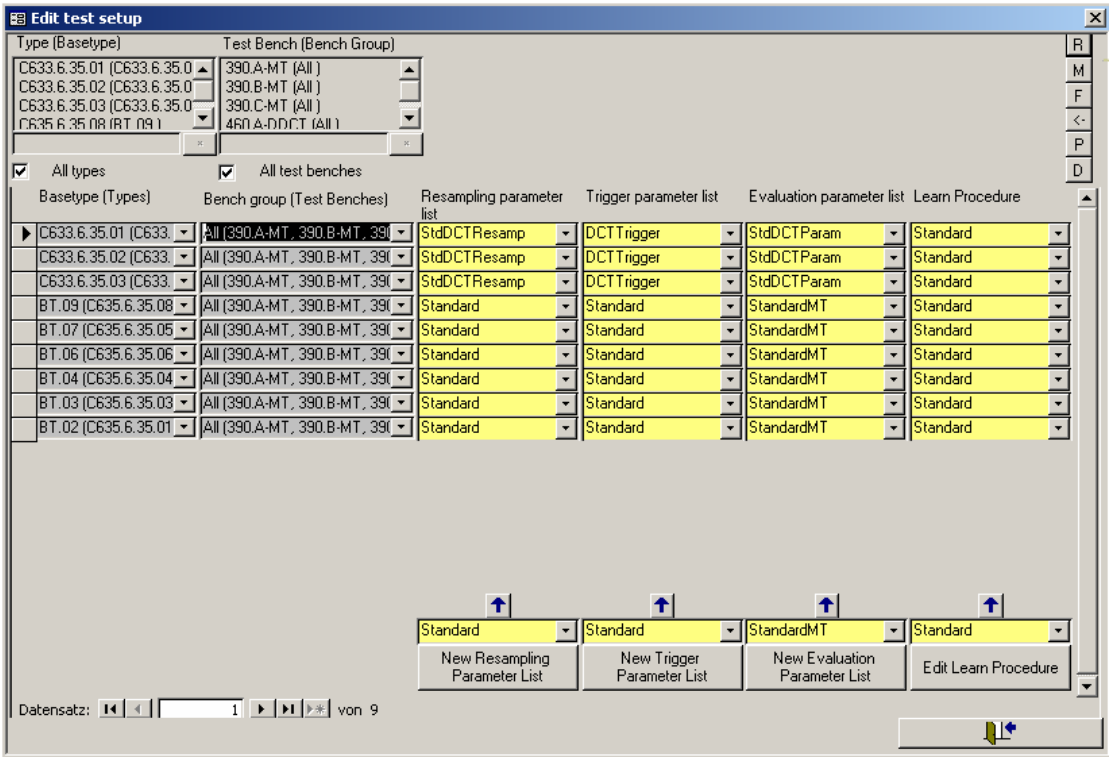
General Design

Data Organization in Lists

Data handling is easy if you follow the rule: „As few data as possible, as much as necessary.“ Generally speaking, this means: Many data-sets which are valid in a wide range of circumstances and few detail data sets. One way to deal with this goal has been explained already as the concept of benches and bench groups or types and base types. Technically speaking, this means that you use an indirect key “base type” or “bench group” to address a data set instead of the direct key “type” or “test bench”. Then you can relate the direct keys “type” and “test bench” to the indirect keys “base type” and “bench group”.

The same idea leads to three important lists in the parameter management: The sampling parameter list, the trigger parameter list and the evaluation parameter list. The basic idea for these lists is that you mostly want to use the same parameters for all types on all test benches, if possible. You add a list for each different set of parameters and relate the list to the types and test benches.

The setup for this task is being done in the following form which can be reached by a click on the button **Test Setup**.



Most of the forms in the parameter management look similar like the one shown above but display different data. The following chapters explain how to read and handle these forms.

General Operation

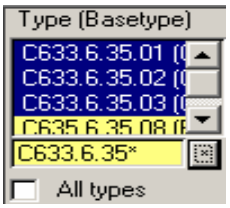
Control area and data area

Each form which is designed like the one above is split in two areas: The *control area* on top of the form which influences what the *data area* displays. On the right side of the control area you find a column with buttons. The function of these buttons is described below. Left of the buttons you see the key selection fields which allow to select some of the possible entries. The corresponding key fields in the data area have gray background color and can not be edited. Fields with yellow background color in the data area contain data which can be edited. The whole form is resizable. This means that if you enlarge the form, the fields in the data area enlarge as well to best use the available space.

Base Types and Bench Groups

As explained above, there is only one data set for all test benches of a bench group as well as for all types of a base type. On the other hand, when you want to access data you have a certain type and a test bench in mind. Thus, to make handling easier, the lists in the key selection fields in the control area have a different content as the corresponding fields in the data area: The key selection fields allow to select a certain type and a test bench (in brackets follows base type or bench group). In contrast, the data area shows the base type or the bench group followed by a list of corresponding types or test benches in brackets. If you select a type, you see in the data area which other types are also affected by a change in this data set.

Using the Key Selection Fields

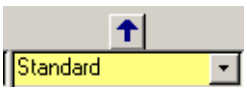


The key selection fields allow to select some entries in different ways. The effect of the check box “All types” is clear at once: If you activate this check box, the other possibilities of the key selection field are deactivated and the data area shows all data sets without any limitation on the corresponding key field.

If you want to select only a few entries, you deactivate the check box „All types“. After that, you can select one or more entries in the list in the typical windows way (select more than one entry by holding the CTRL-key or the Shift-key), or you can use the text field and the button “*” to select entries in the list. In the example above, the Asterisk-button has been used to select all entries which match the pattern “C633.6.35*”. Especially the usual placeholders “*” and “?” can be used (for MS-Access experts: The asterisk-button calls the Access-SQL Function “Like”).

For all buttons, which influence the selection of a selection list keep the following in mind: If the first click does not lead to the expected result, click the button once again. Under certain, at the moment not very clear circumstances, MS Access needs a second “kick”.

Changing all entries of a Column



Often, you want to change the entries in a column for many data sets. The field shown on the right allows to do this. The function is not very difficult: Select something and a click on the arrow-button fills the corresponding column with that value.

In case the column contains numeric values you have the additional function to change values *relatively*. This means, you do not fill the column with one fixed value

but do some calculation with the existing values. You reach this function by entering the desired operation in the field. The following calculations can be done: +X (add the value X), -X (subtract the value X), *X (multiply the value X). In case of the function “-X”, the program asks for the desired function, since „-X“ can mean either to subtract X or to enter the value „-X“.

Sort data sets

Sometimes, it is not enough to select data sets and display them but you want a sorted display. This can be done by clicking any field of a certain column with the right mouse button. Beside other functions, MS Access offers to sort ascending or descending. If you select one of them, the visible data sets are sorted regarding the content of the column where you clicked with the mouse. This function is available for all kind of fields, meaning key fields as well as data fields.

Copy, Print and Compare

On the right side of the control are, you find six buttons as mentioned above. They provide some very effective functions: Copying, printing and comparing data.

The button „P“ stands for „Printing“. It allows to print the current selection in landscape format. Some other buttons are labeled like buttons on a calculator. You might guess the function of the buttons “M” and “R”. “M” allows to memorize a selection, “R” restores a memorized selection (might be necessary to click it twice).

When you click on the button „M“, most forms open another form which can also be reached by clicking the button „F“ (stands for „field selection“). If you want to copy data, you can select the columns which you want to copy, here.

The „<-“ at last starts the copy process in the following way: The data sets of the memorized selection (“M”-button”) are being read and copied to the current selection, if possible.

The difference function, which can be started by clicking the button „D“ works likewise. The data sets of the memorized selection are compared with the data sets of the current selection, if possible. The data area shows the differences. This function must be used carefully, especially if you select great amounts of data or have limited computation speed since the Query which needs to be executed is quite complex.

To indicate that the data area does not show the data of the current selection, the control area changes its background color to brown. In this mode, all differences between the selections are shown.

Another click on the button „D“ results in changing the color to lilac. The color of the letters “M” and “D” change their color as well to indicate that now the data area shows data sets which exist in both selections but have different entries.

If you click the button „D“ once more, the background color and the letter “M” change to pink. Now the data area shows data sets which exist in the memorized selection but not in the current one.

Finally, the situation vice versa as a result of another click on „D“: Now the data area shows data sets which exist in the current selection but not in the memorized one. The color code for that mode is cyan for background color and the letter “D”. You want to click “D” once again? No problem, start again with brown color…

Type management: Create and Remove a Type

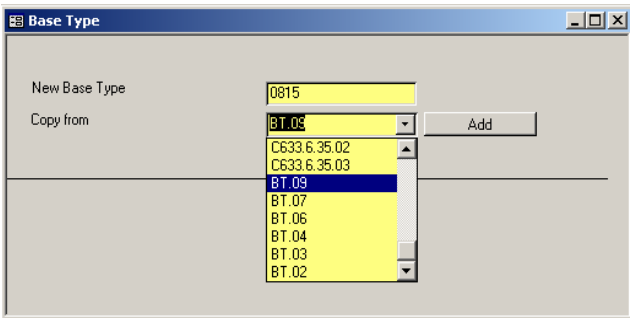
New Type or New Basetype?

When editing parameter for the TAS-System, frequent tasks are the Creation or the Removal of Types. Like mentioned above, already little changes on the transmission housing can lead to a new type identification from the test stand. But it happens also that completely new transmissions for a new vehicle platform are built.

For the database, it is uninteresting why there is a new type. If you want to introduce a new type, at first you need to answer the question whether you need a new base type as well or not. For the measurement system, a new base type is essential when there exists no other base type with exactly the same teeth numbers. Usually, if there exists another base type with the same teeth numbers, you will use that one for the new type and only relate the new type to the existing base type. Under some circumstances, it can be desirable to create a new base type although there exists another one with the same teeth numbers. You will do that if you expect the new type to behave differently concerning noise than the existing base type.

Creating a New Base Type

At first we assume that the new type differs from all existing base types concerning teeth numbers. That means, you have to create a new *Base type*. You reach the corresponding function by clicking the topmost button of the starting form: **Add base type**. The form shown on the right will open:



Enter the new name for the base type in the field „New Base Type“. This name can contain letters and numbers as well. In the example on the right, the new base type shall get the name “0815”.

Since it is not useful to start with a completely empty data set for the new base type, you must specify an existing base type as copy source next. You should select a base type which fits best to the new type. The less differences the two types have, the less you need to change for the new type. To be more exact, at the beginning the new base type will be identical to the existing base type. But it will have its own data set to allow changes.

If you selected an existing base type from the list, click the button **Add**. After that, the new base type is being created and initialized with the data of the source type. At the end of that process, the program returns with the message **Task completed**. Clicking OK closes the message window and the Add Base type – form.

Since a base type alone is not enough to make the data accessible to the measurement program, creating a base type creates a type with the same name as well. (This means, when you create a base type “0815” you create a type “0815” as well and the type is related to that base type.) Since it can happen that the name of the base type already exists as a name for a type, this process can fail. You will then have a new base type with the desired name, but the measurement program can not reach it yet. You need to add another *type* and relate it to the base type to solve that problem.

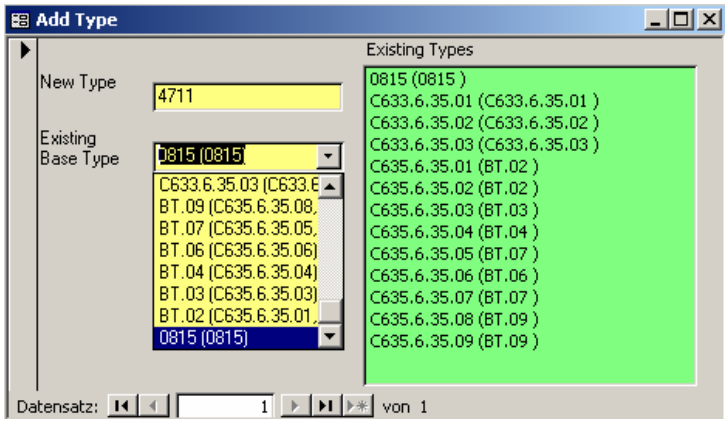
In that context, the creation of a base type can fail for two reasons which lead to corresponding error messages: If you click the button **Add** without having selected an existing base type or if you entered a name which already exist for another base type.

In both cases, it is obvious what to do. You will get no message when you click the button without having entered a name at all. The program will only do nothing, then.

One final remark: The form shown above is not only used for creating a new base type by the program. It is used as well when the task is to create other objects. Behavior and usage are as described above.

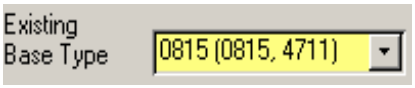
Creating a New Type

If you are so lucky as to use an existing base type for the new type, you only need to relate the new type to the existing base type. You reach this function by clicking the button **Add Type** from the initial form of the parameter management. The following form will open:



The left side with the field and the list box look like those on the form where you create a new base type. Unlike the other form, this form shows a list of the existing types also (showing the corresponding base types in brackets).

The process to create a new type is easy: Enter the name of the new type in the entry field, select the desired base type from the list (again, showing all corresponding types in brackets) and click the button **Add type**. If the task could be completed successfully, the new type shows up in the list of the existing types. In addition to that, the new type has been added to the list in brackets behind the base types in the list box.

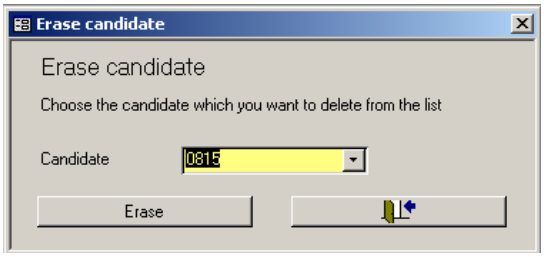


Like the creation of a base type, the creation of a type can fail also if you enter an existing type in the field **New Type**. The program will give an error message if this happens.

Deleting Types and Base Types

Sometimes, transmissions which have been planned are cancelled before the first piece has been built. If you have already created the corresponding types in the database, you want to remove them. You reach this function by clicking the button **Remove Type** from the entry form of the parameter management. The following form will open:

Use the drop down list to select a type from the list (not a base type!). Clicking **Erase** removes this type from the parameter database and has the effect that furthermore, that type is unknown to the measurement program.



In case, the type you want to erase is the last type related to a certain base type, you see the following message:

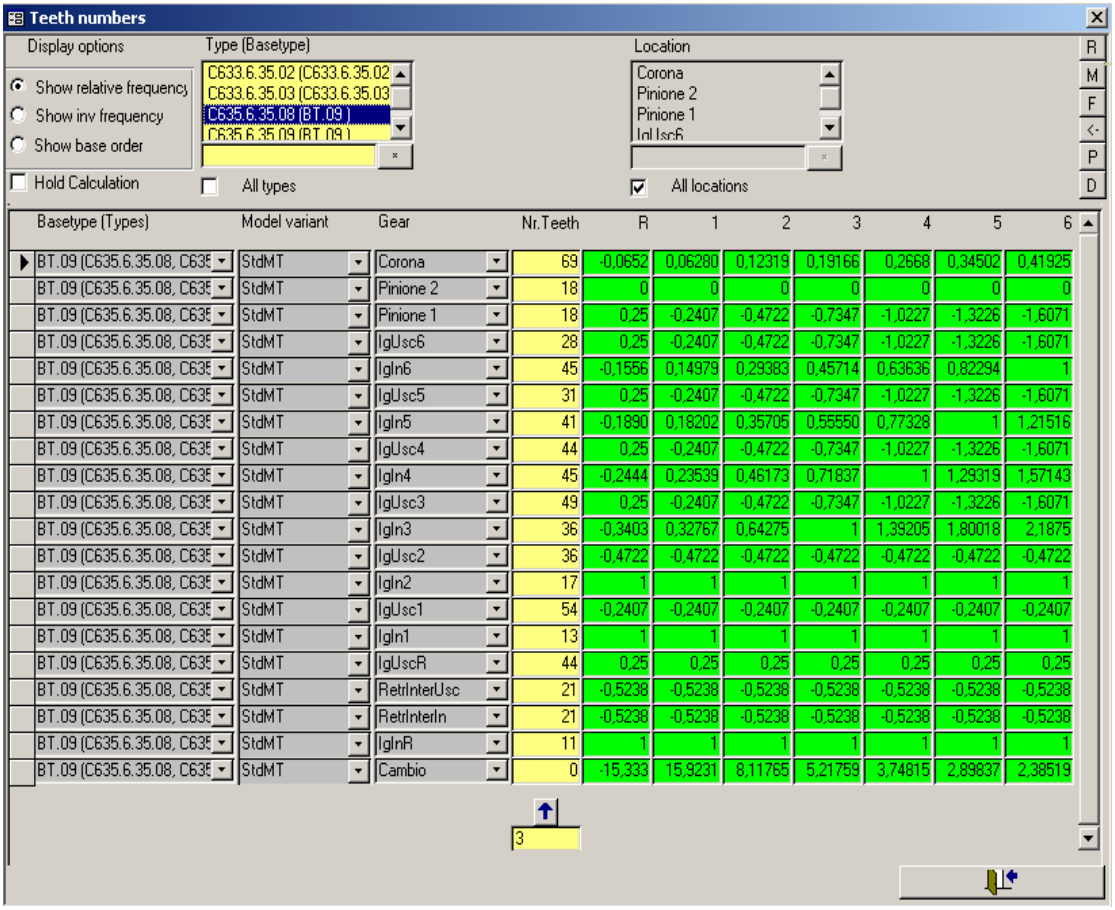
Microsoft Access
Since there is no other candidate name connected (via base number) to this one, you will also delete the base number and all corresponding data.
OK
Abbrechen

If you accept this message clicking **OK**, type *and* corresponding base type are being removed. All settings made for this base type are gone, then (Limits, teeth numbers, etc.) Nothing happens, if you click **Cancel**. Type as well as base type remain in the database.

This procedure is useful since you don't want to hold data in the database which cannot be reached from the measurement program. Like mentioned above below "Creating a base type", this situation can happen. If you want to remove such an "invisible" base type, you need to create and relate a type to it (see above: Creating a new type). Only then you can remove it with the described database function.

Changing Teeth Numbers

The first action, after creating a new base type, is usually to change the teeth numbers. Normally, a new base type has at least partially different teeth numbers. You reach the corresponding form by clicking the button **Construction data** from the initial form of the parameter management. The following form opens:



In this form, the key selection fields mentioned above appear again. Select a type from the list and you will see only the teeth numbers of that type. In the picture above, you see the teeth numbers for type “C635.6.35.08”.

The only column open to make changes is the one with the teeth numbers. The green fields on the right show the relative frequencies of the different wheels in each gear relative to transmission input or transmission output speed as value 1. In the transmission shown above, all frequencies are relative to transmission input speed.

Furthermore, the bottom most row shows the overall ratio of the transmission in each gear. This is useful, since a test bench usually measures this ratio to check the transmission (in case, a wrong wheel was put into the transmission, the ratio wouldn’t match). Sometimes it is also mentioned in the construction lists. If you notice that the ratio shown in the form and the ratio from the test bench or in the construction list do not match, you should check the settings. You may have selected the wrong type or mixed up some teeth numbers (e.g. switching driving and driven wheel). If all teeth numbers are correct, the calculated ratio should match the one on the test bench or in the construction lists exactly.

If you want to check partial ratios, you can switch the display by enabling **Show inv frequency** in the top left corner. In this mode you will see the partial ratios from base speed to each part.

Finally, you can select **Show base orders**. In this mode, the display shows the product of teeth number and relative frequency for each part. These values are in scope of the order analysis. If you have a spectrum with relative frequency 1 to its base speed, the base order is the position in that spectrum where the noise of the specific part will show up. (Later you will read more about positions in the spectrum.)

After explaining the different display options, back to the original task of changing teeth numbers. After a teeth number was changed and the cursor left the corresponding entry field, the parameter management starts to recalculate the relative frequencies of the wheels. Depending on the complexity of the transmission or the performance of the system running the program, this can take a few moments. In case many teeth numbers need changing (maybe for different types also) and this calculation slows down the entering of teeth numbers drastically, you can deactivate the calculation temporarily by enabling the check box **Hold calculation**. You can then enter all teeth numbers quickly. The calculation is done the moment the check box is unchecked again or the moment the form is being closed (please be patient, then).

Test Setup and Learning

Four Lists of Test Parameters

We once took a glance on the form **Edit test setup**, which can be reached by clicking the button **Test Setup** from the entry form of the parameter management. Now I briefly want to explain the parameters behind the lists.

The **Resampling parameter list** defines which *locations* or *rotors* are being sampled with which *sensors*. Only for combinations of locations and sensors which have been defined in this list data blocks can be produced in the measurement program. Moreover, some additional parameter for the sampling process are defined here (length of a data block, number of revolutions per block, etc.). Changes in this list are very rare. If you have the need to change parameters, we advise to contact Discom. If you have more than one transmission variant in the database, you will have at least one Resampling parameter list for each variant.

The **Trigger parameter list** defines the precision for the various measurements. In many situations it also defines the range (see above: Control value) of the measurements. Since the measurement program starts and stops the measurement using the speed information as reference, the start and stop speeds can be found here. You need more than one list if different speed ranges are used for different test stands or different types.

The **Evaluation parameter list** defines which measurements (explanation, see above) shall be measured. Details about how you change settings here, see below. If you have more than one transmission variant in the database, you will again have at least one evaluation parameter list for each variant.

General Settings for the Learning Process

In the list **Learn Procedure** you find the basic settings for the learning process. Beside these basic settings, you can define a lower and an upper bound for each measurement avoiding that the learned limit exceeds a reasonable range. The learned limit will never fall below the lower bound and never climb over the upper bound. If you set the two bounds to the same value, learning is turned off and you have a fixed limit. The learning itself is divided in three phases: Basic training, additional learning and learning complete.

A click on the button **Edit Learn Procedure** opens the following form:

During the basic training, the evaluation is being done with the upper bound as limit because you have too few tests to use a real learned limit. Using the settings shown in the form, this is being done for the first 5 tests of each base type.

After the basic training, learning continues. In this phase, the limit is set on the basis of the learned data but is still being modified. This is being done until the number of measurements defined in **Total number of steps** has been reached. After that the limit is fixed and will not be modified afterwards.

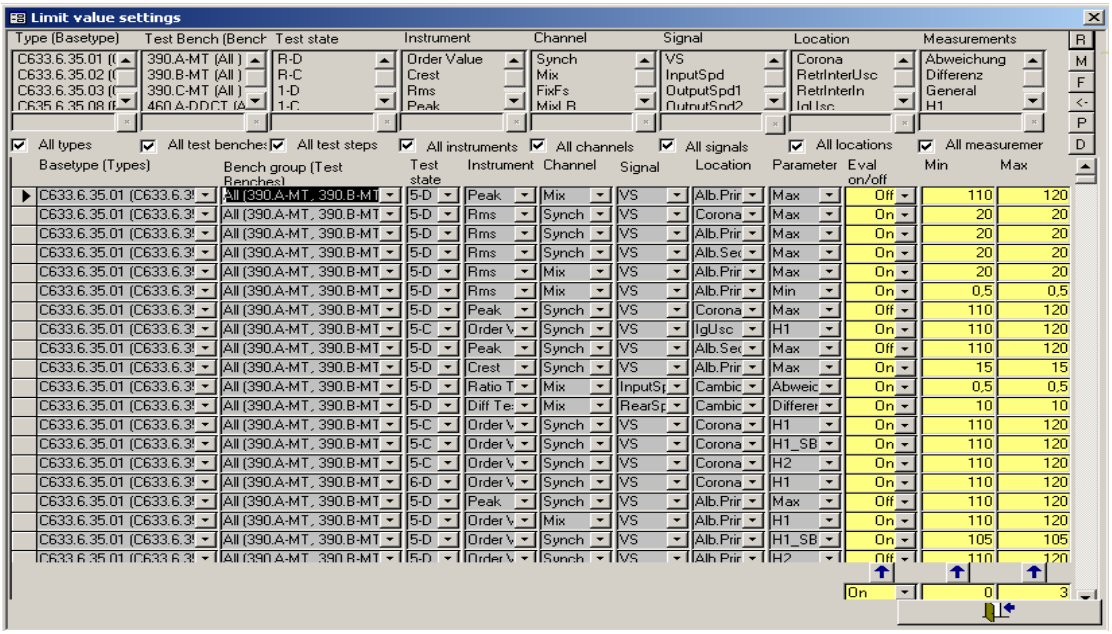
If you want to learn the data for a great number of measurements (e.g. if you want to have a nonstop learning), you often want that newer measurements have greater effect

on the limit than the first measurement thousands of tests before. This behavior is controlled by the **Exp. Time Constant**. The idea is: A value 200 means that each measurement is being treated as if it were the 200st measurement. This allows to follow slight production variances when doing nonstop learning.

Setting Limits

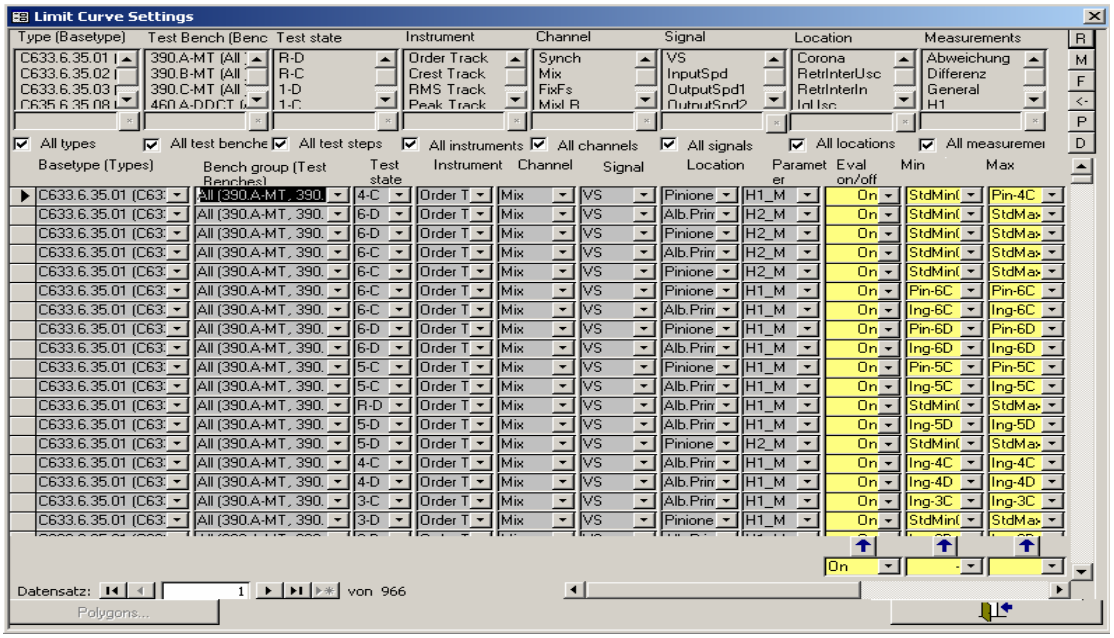
Limits for Single Values

The limit for single values are set in the following form. You reach this form with the button **Single Value Limits** from the initial form of the parameter management. Beside the form for Curve Limits this is the form with the greatest number of key selection fields in the control area.



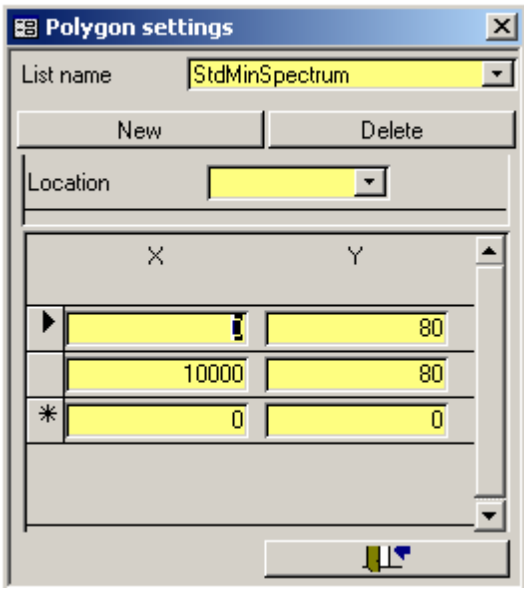
The data you can set here (yellow fields) are whether you want to do an evaluation or not and the bounds for the learning process (see above). If you set the same value for upper and lower bound, the learning is turned off.

Limits for Curve Values



The form for curve value limits looks similar to the one for single value limits. The only difference is that you have to specify polygons instead of single values for the learn boundaries. If you want to create or modify a polygon, you have to select the corresponding instrument in the key selection field first (different instruments often have different value ranges). If you do this, the button **Polygons** in the bottom left corner of the form gets enabled. If you click on that button, the form for managing polygons opens. Like the lists mentioned above, polygons are also defined independently of type and test bench. Polygons have only effect for a measurement when they are used as one of the bounds for a certain type on a certain test bench.

The form on the right shows as an example the settings for a polygon “StdMinSpectrum”. This polygon is defined for the instrument “Order Spectrum”. You read the settings as follows: Each row with X and Y value belong together. The X value defines the order (the X value with the lowest value stands on top of the list).



The polygon in the measurement program is built following this order and connecting each anchor point with the next one using a straight line. In the example, the polygon defines a horizontal line between 0 and 10000 (incl.) with value 80.

We now change the polygon slightly to explain what is being meant with “connecting the anchor points with a straight line”. (X/Y) values noted as pairs: (0/70), (10/80), (10000/80). The upper part of that polygon is again a horizontal line with value 80. The interesting section is between 0 and 10. The anchor points are being connected with a straight line where the value 70 is put at position 0 and the value 80 is

put at position 10. After little thinking, the values for the points in between are clear: Some of these points are: (1/71), (2/72), (5,75) and (9/79).

Please note that due to the fact that the polygon is being ordered by the X value, you can not have two entries for one X-value. Therefore, if you want to define something like a “step” (value 50 left of position 100, value 70 right of position 100), you must enter slightly different X-values, maybe (100/50), (100.01/70).

For a polygon definition the unit for the different values is unknown. The fact that polygons are bound to certain instruments indeed reduce the possibilities but nevertheless, a polygon for a track can have different control values as X-values, maybe Speed, maybe Time or maybe torque, depending which control value is used.

Even the polygons for spectral evaluation can have different references. At first you can have fixed frequency spectra (X-values in Hz) or order spectra (X-values in Ord). Since order spectra always relate to a certain speed reference, you can set a special speed reference for the polygon using a location. If you use this feature, the order positions of the polygon are meant with reference to the speed of that location.

Measurements

Introduction

The analysis system can calculate a variety of standard measurements. If necessary, additional measurements can be added on demand. Nevertheless, most of the typical defects of a transmission, a motor or a single gear on a gear tester should be found with the standard measurements.

The following list shows the standard measurements and explains their relevance for detecting defects.

The Standard Path of Processing

Before we explain the measurements, I want to explain briefly the path of the signals in the measurement system.

At the beginning, the noise signal must be digitally converted using an A/D converter. The A/D converter converts each analogue signal (noise, torque, speed, gear shift force, etc.) with a preset sampling frequency to digital data. Each value of the data mirrors the momentary value on the corresponding connector. Since the A/D converter does its job continually, you get a stream of momentary values for further processing.

The next station on the path of processing is the so called re-sampler. The incoming data stream from the A/D converter is separated into blocks using different criteria. The output of the re-sampler continually produces data blocks instead of a data stream. You often set the re-sampler in a way that the data blocks represent integer multitudes of revolutions of a certain rotor. Like mentioned above this is defined in the sampling parameters. Often, the data blocks from the re-sampler are referred as time signals.

Sync Processing and Mix Processing

A time signal from the re-sampler can be further processed in two different ways: As „Mix“ or as „Sync“.

The major characteristic of the processing path Mix is that the time signal is not manipulated by filters here. This has the effect that the values on the input are mirrored best. Technically speaking this is being done by avoiding averaging in the time domain and using a windowed FFT when converting to the spectral domain.

The Synch processing in contrast does use averaging in the time domain. This has the effect that peaks in the signal are suppressed which do not appear on the same position in the various data blocks. In other words, peaks which do appear on the same positions are clearly detectable after averaging. Technically speaking, non rotor synchronous content is suppressed. The following FFT which does the conversion to the spectral domain in the Sync path is done differently to the mix path also. In the Sync path, the FFT is done without using a window. Once again this has the effect that non rotor-synchronous content is suppressed.

Some applications, especially the gear tester application, process the time signals in the Sync-path with another filter, the so called “Dentist”. This filter removes content from the signal which belong to the teeth. The result is a signal which shows the differences of the teeth more clearly. This filter refines the detection of slight nicks. For spectral evaluation, this filter is useless.

The Unit „Order“

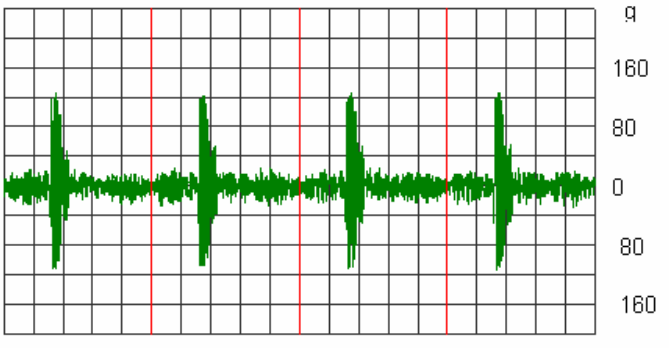
The term order analysis and orders have already been mentioned above. Now it’s time to explain the meaning of order.

If you want to specify frequencies of rotating parts, you have two possible units to relate to: Hertz and Order. Both units are defined using a sinus signal. A sinus signal has the frequency of 1 Hz if there is exactly one peak within 1 sec.

If you have rotating parts, the unit Hertz may not be the best choice. If the speed varies, the noise frequency which the parts produce varies also: Double frequency for double speed. If you want to compare both noise situations, you need a reference which is speed independent. The Order is this speed independent reference. A sinus signal has the frequency of 1 Ord if there is exactly one peak within 1 *revolution*.

The Measurement Peak

The figure on the right shows a signal in the Synch processing channel after averaging in the time domain.



The re-sampler has been set to put 4 revolutions in one data block which has been emphasized by adding red lines. You clearly see the high values in the spectra for each revolution which are caused by a nick on the corresponding rotor. The measurement peak captures this high peak directly. The analysis program would produce a peak value around 100 from the signal shown above.

The Measurement Rms

Not only the peak value of a signal is relevant but also the overall energy it contains. The measurement Rms gives information about that. The value is being calculated for each data block by squaring each value, then calculating the average value over the result block and at the end calculating the square root of the average value. This explains the abbreviation RMS = “Root Mean Squared”.

The Measurement Crest

If you want to find a nick, the peak value often is not enough. It can happen that two test parts show different peak values resulting from a loud overall noise of one part, whereas the other one has a clearly visible nick (meaning that there is a visible peak in the time signal). The measurement Crest allows to find this nick.

The value Crest is calculated by taking the Peak and the Rms value of a data block and calculating the quotient of both. As a formula: Crest = Peak/Rms.

If there is a high Peak, but a low RMS value, the Crest will produce a high value also, whereas if there is a high Peak and a high RMS value, the Crest will produce a low value. If you have multiple nicks on a part, this fact makes detection difficult since a signal with many peaks will also produce a high RMS value.

The Measurement Kurtosis

The measurement Kurtosis gives a measure for the „Peakliness“ of a signal. A clear sinus signal produces a Kurtosis value of 1. The value gets higher the more Peaks the signal has. This qualifies the Kurtosis as an indicator for multiple nicks.

The Measurement Spectrum

When describing the different processing paths, it was already mentioned that the time signal is being transformed into the spectral domain by doing a FFT. This abbreviation stands for “Fast Fourier Transformation”. (The mathematical procedure Fourier Transformation gets fast if it is being done on data blocks with a length that is a power of 2. For this reason, only such block lengths are allowed in the sampling parameters.) The result of the Fourier Transformation is a spectrum. If you transform a data block containing a sinus signal which fits to the block length, the spectrum shows only one peak at the position which corresponds to the frequency of the sinus signal. Furthermore, the peak shows the amplitude of the sinus signal. Looking vice versa: If you have a spectrum, you can get a fitting time signal by overloading a number of sinus signals. The amplitude and frequency of the sinus signals are defined by position and amplitude of the spectral line.

To summarize: The spectrum is a representation of the sound of a signal. In contrast to the other measurements described before (which were single values), the spectrum is a curve. If you use this measurement with learned limit values for evaluation, you can separate test parts which sound remarkably different than other parts.

The Measurement Spectral Value

Generally speaking, the spectrum shows the sound of a signal. Moreover, some positions in the spectrum, especially if you have an order spectrum, have special meaning and give important information about a specific part.

As an example, I take a look at a gear wheel with 38 teeth. The noise it generates is surely no sinus signal, but nevertheless you expect that each teeth of the gear wheel is responsible for a peak in the time signal. Consequently, the order spectra shows a peak at order 38. This order position is called the “Gear mesh order”. Beside this order, the integer multitudes of this order are also relevant (the so called harmonics). Normally, meshing gears are the major noise source in a transmission which explains the importance of these orders. If a transmission is classified as loud by a human, you often find high amplitudes for the gear mesh harmonic orders in the spectrum. You can measure values at special spectral positions separately by a measurement called “Spectral value”.

Not only the gear mesh order harmonics are important, the area directly beside these orders is also important. The so called *side bands* show high amplitudes if the gear wheel runs eccentric.

Finally, sometimes it happens that order positions have high amplitude which correspond to fractions of the gear mesh order (for the gear wheel with 38 teeth this could be order 16 = half gear mesh). Damaged or worn out grinding wheels in the production can produce such defects.

Track Measurements

The measurements so far have one thing in common: The values which are taken from the single data blocks are being maximized, minimized or averaged during the measure time (depending on the settings in the database) and produce a final value (e.g. a spectrum). With this procedure you miss the information what happens during the measurement when the test stand varies speed or torque. Sometimes, transmissions show their defects only in special speed/torque conditions and cannot be detected if you only look at the summarized values.

This gap can be filled by using the different track measurements for Peak, Rms, Crest, Kurtosis, Spectral Value and Spectra. They allow to capture and evaluate the behavior of a value with respect to a control value in a curve. If you track curves, you

get a spectrogram. Spectrograms are, measured by the amount of data they create as result, the largest measurement but also the most exact one.

Measurements which are Calculated at the End of a Test

Most of the measurements described above are determined directly by averaging, maximizing, minimizing or recording during the measure time. The only exception so far is the spectral value since this value can only be determined after the calculation of the corresponding spectrum has finished.

Beside the spectral value there are other measurements which can be calculated only after other measurements. Some of them are the curve interval and the curve polygon.

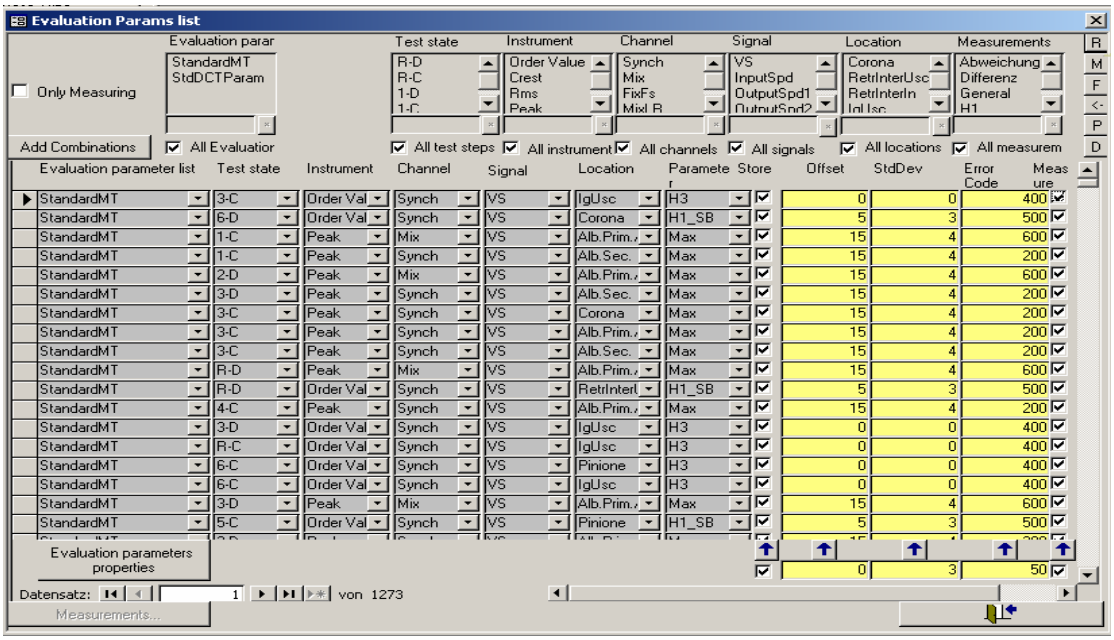
Both evaluations need a curve to work on as input. The curve interval calculates a value for a section of this curve (maximizing, minimizing or averaging the values of this section). If you do this for a tracking curve, you can simplify the analysis of a tracking curve from curve analysis to single value analysis. If you set the section wisely, these single values represent the characteristics of the whole curve. The advantage of this simplification is that you can easily do statistics for these values (like for all single values).

The curve polygon is used to compare a curve with a polygon and calculate a characteristic value for the curve. The easiest thing is to determine minimum or maximum within the value interval of the polygon (like the curve interval) but the specialty of the curve polygon is to calculate the area between curve and polygon (either above the curve or below the curve). This kind of evaluation is being done for the analysis of curves which show the gear switch force relative to distance. The calculated value represents the gear shift work.

Specifying Measurements

General Measurement Parameters

The following form allows to set the general measurement parameters. It opens when you expand the initial form by enabling the check box **Show advanced settings** and click the button **Measurement Value Setup**.



Like mentioned above this list is not dependant of types and test benches. You can define different lists and use them for the different types or test benches.

The *List of measurements* defines the following parameters for the different measurements: The error code which is reported if the evaluation is not ok; whether the measurement shall be measured or not (this allows to disable measurements without having to delete them); and whether the measurement shall be stored in the result data store or not. (Disabling storage makes sense for measurements which are only used to be the source for other calculations.) Furthermore you can set two parameters **Offset** and **StdDev** which influence the calculation of learned limits heavily.

It was already mentioned above that the limit which is calculated from average and standard deviation keeps its value between lower and upper bound set in the limit settings. The exact calculation formula for the limit is as follows: *Average* + **Offset** + **StdDev** * *Standard Deviation*. Average and Standard Deviation are the values which are learned during many test runs, Offset and StdDev are the parameters which can be set in the list of measurements. Practically speaking, these parameters work as follows: With the *Offset* value you can move a learned limit. If the calculated limit value lies too tight to the measured values and you get unwanted nok results, you can change the offset value and move the limit out of the critical area. On the other hand, with the *StdDev* value you move the limit curve out of a critical area if you have great differences between single measurements. A high StdDev value results in more distance if there is much difference between the single measurements.

Both settings have to be used carefully since generally they are valid for many types and test benches. Changing one parameter results in a change of this parameter for all types and test benches where the corresponding list is being used!

Adding Measurements

The form for measurement value setup is the first form in this manual which has an additional button on the left of the key selection fields labeled **Add Combinations**. The function of this button is easily guessed: Adding entries to the list of measurements. Again, this button must be used carefully, since it can fill the database with many unwanted entries!

If you want to add entries, you have to select these keys in the key selection fields which you miss in the list of measurements. For the beginner I suggest to start by disabling **all** check boxes which enable all keys (e.g. “all test steps”). Then start on the left hand side and enable just those entries in the key selection fields which you want to add. Furthermore, I suggest to use multiple selections (or select all entries) only one key list and avoid multiple selections in other lists. That way, you avoid filling the database with unwanted entries.

Let me explain, how easily wrong entries can be added to the list in an example: Assume, you have added a new measurement parameter H5 for instrument Spectral value. You want to have this new measurement for three locations GearIn, GearOut and Reverse Wheel in all fitting test steps for both sensors (S1 and S2). If you are not careful, you may select the following: “All test steps”, Instrument: Spectral value, “All Channels”, “All Signals”, Location: GearIn, GearOut and Reverse Wheel, Parameter: H5. Clicking “Add Combinations” fills the list with all possible combinations of these entries and result in especially the following unwanted entries:

- Unwanted entries for the key „Sensor“. For some measurements, speed and torque are correct entries. Therefore, they are contained in the key selection list for the sensor. Nevertheless, you mostly use noise sensors. Therefore, if you want to *display* both sensors, you usually select “All sensors”. When adding entries, “All sensors “means indeed *all* sensors, including speed and torque which does not make sense in certain combinations.
- Unwanted entries for the key location depending on the test step. The reason is the following: For a manual gearbox GearIn and GearOut usually represent the different gear wheels for each gear which carry torque. In between these two gears, the reverse wheel is located but it carries torque only in the test steps for the reverse gear. Usually, it does not make sense to measure this wheel in other test steps than those for the reverse gear.

Deleting Measurements

If you find entries in the list of measurements which are unwanted, maybe because you were clicking the button „Add combinations“ too fast, you can also delete them. Please note, that left of the first column of the data area (beside the list name) you find a triangle. This is the so called “Data set selector”. It marks that data set in the visible form which is selected for editing. You can mark rows using this first column using the shift key (typical windows-like). The column with the data set selector will be marked in reverse for the selected rows. If you press the Del key on the keyboard, the parameter management will delete the marked entries form the database completely.

Interaction between List of Measurements and Limit Settings

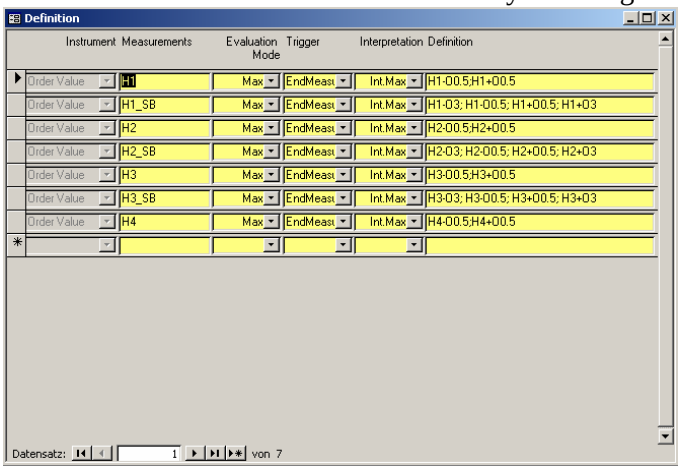
Since you don't want to add the same measurements in different forms, the parameter management assumes that each measurement which is being measured shall also be evaluated. Consequently, if you add measurements to the list of measurements, you also add the same entries for the corresponding limits. This process is quite fast. When deleting measurements from the list of measurements, the same is being done: Useless entries are removed from the limit settings either, but this is a process which takes some more time. You may need to be patient until the parameter management has finished this task.

A result of spreading parameter entries between these forms is that you may have entries in the limit lists which are not being measured. Example: You have two lists of measurements "List1" and "List2". List1 uses a measurement H5, List2 uses a measurement "H5_SB" instead. Both lists are used for at least one type/test bench. In effect, you find limit entries for both measurements for all types and test benches.

The reason for this is follows: Assume, a type/test bench is using List1 and its limits are finely set. Now, you want to use List2 for a special test run. After this special test run you will still use the old limits. The only way for the parameter management to ensure this, is to keep all possible limit entries in existence, disregarding which list is currently used.

Adding New Measurements

After explaining how you get new measurements into the list of measurements, I still need to explain how to *define* a new measurement at all. We do this by selecting the instrument "Spectral value" (and just this one) in the corresponding key selection list. After doing this, the button **Measurements** in the bottom left hand corner is being enabled. A click on this button opens the form shown on the right:



I skip the description of the syntax for the definition entry here. I only suggest to refer to the existing entries.

You can add the newly defined measurement in the bottom row of the form (beside the *). Please note that the program verifies that all fields have been filled out before you leave a row (otherwise you get an error message). Therefore, if you want to copy and change an existing definition, you should prepare this *before* you add entries. Otherwise you must enter anything and correct the definition afterwards.

You can add a measurement to the list of measurements only after you defined it here. (To be more exact: You define the parameter for a measurement.)

For other instruments similar forms exist to define new parameters. The specialties depend on the function of the measurement instrument itself (see above: Description of the measurements).

Security and Maintainance Issues

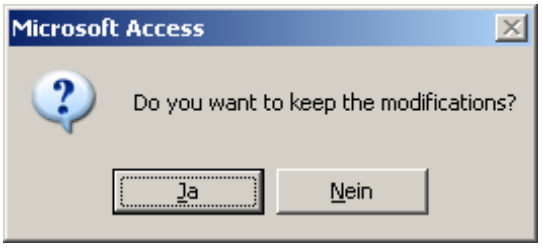
Database-Backup

If you click the wrong button while changing entries in the database, many entries can become useless. It can be impossible or at least a time consuming task to recreate these entries. Therefore you may ask for an Undo-button to get back to the original state. The parameter management system does not have such a button, but nevertheless it provides a helpful function for this situation.

If you start the parameter management, the first thing it does is to create a backup of the current data files. This means that unwanted changes can always be undone by restoring the “old” database file.

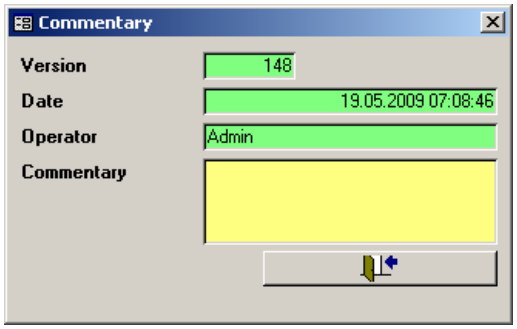
If the parameter management notices changes it prompts with the following message when closing the program:

Clicking „No“ forces the parameter management system to restore the last database backup (the one created at startup) and to discard the current database file. Clicking “Yes” keeps all entries as they currently are.



Please note that every change in the parameters is activated *at once*. If the cacher queries for changed data while data sets are being changed this can even result in temporarily inconsistent cache data files.

If you prompt the message above with „yes“, the form shown on the right opens. With this form you can give a commentary to the changes you have made. You can reach this form by clicking the button Commentary on the intital form. There you can not make an entry but can navigate through the commentaries and see which changes have been done to the parameters.



De-fragmenting the database

After having changed many entries in the database (especially if you deleted some entries), I suggest to de-fragment the database. Deleting entries always result in a fragmented database. To be more exact, Access does the following: Assume, the database contains Data sets 1-5 ordered as follows: Data set 1, Data set 2, Data set 3, Data set 4, Data set 5. If you delete the data sets 3 and 4, Access does not automatically move Data set 5 to the end of Data set 2, but keeps the area unused which was occupied by data set 3 and 4. It is not sure, whether Access uses this area otherwise or not.

A fragmented data base file uses more space on the hard disk than it really needs. Accessing data sets can be slowed down as well. Thus, de-fragment the database file after deleting much data. You find the corresponding button in the **Advanced settings** section of the initial form.

External Database Modifications

You may want to make modifications of the database on another system. This may be a server or you request Discom to modify your database. For that you need to know how to retrieve and substitute the actual MS Access database file.

The parameter database file is located in

C:\Discom\Measurements\MultiRot\Project\ParamDb.

The project 'Project' has to be substituted with your specific application name. Inside this directory, you find a MS access file, again with a project specific name like Project_QDb.mdb. This file stores all your settings. If you may mail this file to Discom for assistance please zip it. This both makes the file much smaller and also allows mail programs to retrieve the content. Files with the extension .mdb are normally blocked in Outlook.

If you receive a modified database, please perform the following step:

1) Make a backup of the existing database
2) Stop the measurement program.
3) Put the new database in place. Remember that you may have to unzip the mail attachment.
4) After you have substituted the database file please delete all the files in the folder *C:\Discom\Measurements\MultiRot\Project\LocalsCacheData/*.sea.*
   Delete the content of the folder, not the folder itself. This makes sure that Rotas will use the new database and not some old cached version of the parameters.
5) Start the Rotas measurement program again.